


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☐ The ACM Digital Library ☒ The Guide


THE GUIDE TO COMPUTING LITERATURE

 [Feedback](#)

## Concurrent garbage collection using hardware-assisted profiling

 Full text  [Pdf \(1.74 MB\)](#)

Source [International Symposium on Memory Management](#) [archive](#)  
[Proceedings of the 2nd international symposium on Memory management](#) [table of contents](#)  
 Minneapolis, Minnesota, United States  
 Pages: 80 - 93  
 Year of Publication: 2000  
 ISBN:1-58113-263-8  
[Also published in ...](#)

Authors [Timothy H. Heil](#) Electrical and Computer Engineering, University of Wisconsin - Madison, 1415 Engineering Drive  
[James E. Smith](#) Electrical and Computer Engineering, University of Wisconsin - Madison, 1415 Engineering Drive

 Sponsor [SIGPLAN: ACM Special Interest Group on Programming Languages](#)

 Publisher [ACM](#) New York, NY, USA

Bibliometrics Downloads (6 Weeks): 4, Downloads (12 Months): 28, Citation Count: 5

 Additional Information: [abstract](#) [cited by](#) [index terms](#) [collaborative colleagues](#) [peer to peer](#)

Tools and Actions: [Review this Article](#)  
[Save this Article to a Binder](#) Display Formats: [BibTex](#) [EndNote](#) [ACM Ref](#)

DOI Bookmark: Use this link to bookmark this Article: <http://doi.acm.org/10.1145/362422.362466>  
[What is a DOI?](#)

### ⤴ ABSTRACT

In the presence of on-chip multithreading, a Virtual Machine (VM) implementation can readily take advantage of *service threads* for enhancing performance by performing tasks such as profile collection and analysis, dynamic optimization, and garbage collection concurrently with program execution. In this context, a hardware-assisted profiling mechanism is proposed. The *Relational Profiling Architecture* (RPA) is designed from the top down. RPA is based on a relational model similar to the relational database model. Instructions selected for profiling produce a record of information. A simple *query engine* examines these records for patterns, and performs simple actions on matching records.

The power and flexibility of RPA is demonstrated by developing a concurrent generational garbage collector for Java. Detailed execution driven simulations show that this collector has an average runtime overhead of approximately 0.6%. The short pauses in the application required for synchronization with the garbage collector are at most 54 microseconds, given a 1GHz clock frequency.

### ⤴ CITED BY 5



[Timothy Heil, James E. Smith, Relational profiling: enabling thread-level parallelism in virtual machines, Proceedings of the 33rd annual ACM/IEEE international symposium on Microarchitecture, p.281-290, December 2000, Monterey, California, United States](#)



G. Chen , R. Shetty , M. Kandemir , N. Vijaykrishnan , M. J. Irwin , M. Wolczko, Tuning garbage collection for reducing memory system energy in an embedded java environment, ACM Transactions on Embedded Computing Systems (TECS), v.1 n.1, p.27-55, November 2002

David Detlefs , Ross Knippel , William D. Clinger , Matthias Jacob, Concurrent Remembered Set Refinement in Generational Garbage Collection, Proceedings of the 2nd Java™ Virtual Machine Research and Technology Symposium, p.13-26, August 01-02, 2002



Cliff Click , Gil Tene , Michael Wolf, The pauseless GC algorithm, Proceedings of the 1st ACM/USENIX international conference on Virtual execution environments, June 11-12, 2005, Chicago, IL, USA



Ajeet Shankar , S. Subramanya Sastry , Rastislav Bodik , James E. Smith, Runtime specialization with optimistic heap analysis, ACM SIGPLAN Notices, v.40 n.10, October 2005

## ↑ INDEX TERMS

### Primary Classification:

D. Software



D.4 OPERATING SYSTEMS



D.4.2 Storage Management



Subjects: Garbage collection

### Additional Classification:

D. Software



D.3 PROGRAMMING LANGUAGES



D.3.2 Language Classifications



Subjects: Macro and assembly languages



D.3.4 Processors



Subjects: Memory management (garbage collection); Compilers



D.4 OPERATING SYSTEMS



D.4.1 Process Management



Subjects: Concurrency; Threads

F. Theory of Computation



F.1 COMPUTATION BY ABSTRACT DEVICES



F.1.2 Modes of Computation



Subjects: Parallelism and concurrency

K. Computing Milieux



K.6 MANAGEMENT OF COMPUTING AND INFORMATION SYSTEMS



K.6.2 Installation Management



Subjects: Benchmarks

### General Terms:

Design, Management, Performance, Theory

↗ Collaborative Colleagues:

Timothy H. Heil: [colleagues](#)

James E. Smith: [colleagues](#)

↗ Peer to Peer - Readers of this Article have also read:

- [Data structures for quadtree approximation and compression](#) Communications of the ACM 28, 9  
Hanan Samet
- [A hierarchical single-key-lock access control using the Chinese remainder theorem](#) Proceedings of the 1992 ACM/ SIGAPP Symposium on Applied computing  
Kim S. Lee , Huizhu Lu , D. D. Fisher
- [An intelligent component database for behavioral synthesis](#) Proceedings of the 27th ACM/ IEEE conference on Design automation  
Gwo-Dong Chen , Daniel D. Gajski
- [The GemStone object database management system](#) Communications of the ACM 34, 10  
Paul Butterworth , Allen Otis , Jacob Stein
- [Putting innovation to work: adoption strategies for multimedia communication systems](#) Communications of the ACM 34, 12  
Ellen Francik , Susan Ehrlich Rudman , Donna Cooper , Stephen Levine

↗ This Article has also been published in:

- [ACM SIGPLAN Notices](#)  
Volume 36 , Issue 1 , Jan 2001

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2008 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)